

IN THE CLAIMS

This listing of claims replaces all prior listings:

1-15. (Canceled)

16. (Previously Presented) A method in a data processing system for remote inter-language method calling comprising:

receiving, at a connect module, a method call from a proxy object using an intermediary protocol, the method call being translated from a first protocol to the intermediary protocol by the proxy object;

translating the method call, by the connect module, from the intermediary protocol to a second protocol, wherein the connect module includes stub implementations for a plurality of protocols and a call parameter data structure for storing call parameters and return values; and

issuing the method call, by the connect module, to a method using said second protocol.

17. (Previously Presented) The method of claim 16 wherein said first protocol is selected from the group of Java, XPCOM and UNO.

18. (Previously Presented) The method of claim 16 wherein said second protocol is selected from the group of Java, XPCOM and UNO.

19. (Previously Presented) The method of claim 16 wherein the plurality of protocols include Java, XPCOM and UNO.

20. (Previously Presented) A remote inter-language method calling system comprising:

a memory having a program that receives, at a connect module, a method call from a proxy object using an intermediary protocol, the method call being translated from a first protocol to the intermediary protocol by the proxy object; translates the method call, at the connect module, from the intermediary protocol to a second protocol, wherein the connect

module includes stub implementations for a plurality of protocols and a call parameter data structure for storing call parameters and return values; and issues the method call, from the connect module, to a method using said second protocol; and  
a processor that runs the program.

21. (Previously Presented) The remote inter-language method calling system of claim 20 wherein said first protocol is selected from the group of Java, XPCOM and UNO.

22. (Previously Presented) The remote inter-language method calling system of claim 20 wherein said second protocol is selected from the group of Java, XPCOM and UNO.

23. (Previously Presented) The remote inter-language method calling system of claim 20 wherein the plurality of protocols include Java, XPCOM and UNO.

24. (Previously Presented) A tangible computer-readable medium having computer-readable program code embodied therein configured for remote inter-language method calling, comprising:

computer readable code configured to cause said computer to receive, at a connect module, a method call from a proxy object using an intermediary protocol, the method call being translated from a first protocol to the intermediary protocol by the proxy object;

computer readable code configured to cause said computer to translate the method call, by the connect module, from the intermediary protocol to a second protocol, wherein the connect module includes stub implementations for a plurality of protocols and a call parameter data structure for storing call parameters and return values; and

computer readable code configured to cause said computer to issue the method call, by the connect module, to a method using said second protocol.

25. (Previously Presented) The computer-readable medium of claim 24 wherein said first protocol is selected from the group of Java, XPCOM and UNO.

26. (Previously Presented) The computer-readable medium of claim 24 wherein said second protocol is selected from the group of Java, XPCOM and UNO.

27. (Previously Presented) The computer-readable medium of claim 24 wherein the plurality of protocols include Java, XPCOM and UNO.

28. (Previously Presented) The method of claim 16 wherein the call parameters are placed by value in the call parameter data structure.

29. (Previously Presented) The remote inter-language method calling system of claim 20 wherein the call parameters are placed by value in the call parameter data structure.

30. (Previously Presented) The computer-readable medium of claim 24 wherein the call parameters are placed by value in the call parameter data structure.